



# OpenShift Dedicated 4

## Security and compliance

Configuring security context constraints in OpenShift Dedicated



## OpenShift Dedicated 4 Security and compliance

---

Configuring security context constraints in OpenShift Dedicated

## Legal Notice

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This document provides instructions for configuring security context constraints in OpenShift Dedicated.

---

## Table of Contents

<b>CHAPTER 1. VIEWING AUDIT LOGS</b> .....	<b>3</b>
1.1. ABOUT THE API AUDIT LOG	3
1.2. VIEWING THE AUDIT LOGS	4
1.3. FILTERING AUDIT LOGS	8
1.4. GATHERING AUDIT LOGS	9
1.5. ADDITIONAL RESOURCES	10
<b>CHAPTER 2. REQUIRED ALLOWLIST IP ADDRESSES FOR SRE CLUSTER ACCESS</b> .....	<b>11</b>
2.1. OVERVIEW	11
2.2. OBTAINING ALLOWLISTED IP ADDRESSES	11



# CHAPTER 1. VIEWING AUDIT LOGS

OpenShift Dedicated auditing provides a security-relevant chronological set of records documenting the sequence of activities that have affected the system by individual users, administrators, or other components of the system.

## 1.1. ABOUT THE API AUDIT LOG

Audit works at the API server level, logging all requests coming to the server. Each audit log contains the following information:

Table 1.1. Audit log fields

Field	Description
<b>level</b>	The audit level at which the event was generated.
<b>auditID</b>	A unique audit ID, generated for each request.
<b>stage</b>	The stage of the request handling when this event instance was generated.
<b>requestURI</b>	The request URI as sent by the client to a server.
<b>verb</b>	The Kubernetes verb associated with the request. For non-resource requests, this is the lowercase HTTP method.
<b>user</b>	The authenticated user information.
<b>impersonatedUser</b>	Optional. The impersonated user information, if the request is impersonating another user.
<b>sourceIPs</b>	Optional. The source IPs, from where the request originated and any intermediate proxies.
<b>userAgent</b>	Optional. The user agent string reported by the client. Note that the user agent is provided by the client, and must not be trusted.
<b>objectRef</b>	Optional. The object reference this request is targeted at. This does not apply for <b>List</b> -type requests, or non-resource requests.
<b>responseStatus</b>	Optional. The response status, populated even when the <b>ResponseObject</b> is not a <b>Status</b> type. For successful responses, this will only include the code. For non-status type error responses, this will be auto-populated with the error message.

Field	Description
<b>requestObject</b>	Optional. The API object from the request, in JSON format. The <b>RequestObject</b> is recorded as is in the request (possibly re-encoded as JSON), prior to version conversion, defaulting, admission or merging. It is an external versioned object type, and might not be a valid object on its own. This is omitted for non-resource requests and is only logged at request level and higher.
<b>responseObject</b>	Optional. The API object returned in the response, in JSON format. The <b>ResponseObject</b> is recorded after conversion to the external type, and serialized as JSON. This is omitted for non-resource requests and is only logged at response level.
<b>requestReceivedTimestamp</b>	The time that the request reached the API server.
<b>stageTimestamp</b>	The time that the request reached the current audit stage.
<b>annotations</b>	Optional. An unstructured key value map stored with an audit event that may be set by plugins invoked in the request serving chain, including authentication, authorization and admission plugins. Note that these annotations are for the audit event, and do not correspond to the <b>metadata.annotations</b> of the submitted object. Keys should uniquely identify the informing component to avoid name collisions, for example <b>podsecuritypolicy.admission.k8s.io/policy</b> . Values should be short. Annotations are included in the metadata level.

Example output for the Kubernetes API server:

```
{
  "kind": "Event",
  "apiVersion": "audit.k8s.io/v1",
  "level": "Metadata",
  "auditID": "ad209ce1-fec7-4130-8192-c4cc63f1d8cd",
  "stage": "ResponseComplete",
  "requestURI": "/api/v1/namespaces/openshift-kube-controller-manager/configmaps/cert-recovery-controller-lock?timeout=35s",
  "verb": "update",
  "user": {
    "username": "system:serviceaccount:openshift-kube-controller-manager:localhost-recovery-client",
    "uid": "dd4997e3-d565-4e37-80f8-7fc122ccd785",
    "groups": [
      "system:serviceaccounts",
      "system:serviceaccounts:openshift-kube-controller-manager",
      "system:authenticated"
    ],
    "sourceIPs": ["::1"],
    "userAgent": "cluster-kube-controller-manager-operator/v0.0.0 (linux/amd64) kubernetes/$Format",
    "objectRef": {
      "resource": "configmaps",
      "namespace": "openshift-kube-controller-manager",
      "name": "cert-recovery-controller-lock",
      "uid": "5c57190b-6993-425d-8101-8337e48c7548",
      "apiVersion": "v1",
      "resourceVersion": "574307"
    },
    "responseStatus": {
      "metadata": {},
      "code": 200
    },
    "requestReceivedTimestamp": "2020-04-02T08:27:20.200962Z",
    "stageTimestamp": "2020-04-02T08:27:20.206710Z",
    "annotations": {
      "authorization.k8s.io/decision": "allow",
      "authorization.k8s.io/reason": "RBAC: allowed by ClusterRoleBinding 'system:openshift:operator:kube-controller-manager-recovery' of ClusterRole 'cluster-admin' to ServiceAccount 'localhost-recovery-client/openshift-kube-controller-manager'"
    }
  }
}
```

## 1.2. VIEWING THE AUDIT LOGS



You can view the logs for the OpenShift API server, Kubernetes API server, OpenShift OAuth API server, and OpenShift OAuth server for each control plane node.



## NOTE

In OpenShift Dedicated deployments, customers who are not using the Customer Cloud Subscription (CCS) model must request a copy of your cluster's audit logs by contacting Red Hat Support. This is because viewing API server audit logs requires **cluster-admin** privileges.

## Procedure

To view the audit logs:

- View the OpenShift API server audit logs:
  - a. List the OpenShift API server audit logs that are available for each control plane node:

```
$ oc adm node-logs --role=master --path=openshift-apiserver/
```

### Example output

```
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit-2021-03-09T00-12-19.834.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit-2021-03-09T00-11-49.835.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit-2021-03-09T00-13-00.128.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit.log
```

- b. View a specific OpenShift API server audit log by providing the node name and the log name:

```
$ oc adm node-logs <node_name> --path=openshift-apiserver/<log_name>
```

For example:

```
$ oc adm node-logs ci-ln-m0wpfjb-f76d1-vnb5x-master-0 --path=openshift-apiserver/audit-2021-03-09T00-12-19.834.log
```

### Example output

```
{"kind":"Event","apiVersion":"audit.k8s.io/v1","level":"Metadata","auditID":"381acf6d-5f30-4c7d-8175-c9c317ae5893","stage":"ResponseComplete","requestURI":"/metrics","verb":"get","user":{"username":"system:serviceaccount:openshift-monitoring:prometheus-k8s","uid":"825b60a0-3976-4861-a342-3b2b561e8f82","groups":["system:serviceaccounts","system:serviceaccounts:openshift-monitoring","system:authenticated"]},"sourceIPs":["10.129.2.6"],"userAgent":"Prometheus/2.23.0","responseStatus":{"metadata":{},"code":200},"requestReceivedTimestamp":"2021-03-08T18:02:04.086545Z","stageTimestamp":"2021-03-08T18:02:04.107102Z","annotations":
```

```
{ "authorization.k8s.io/decision": "allow", "authorization.k8s.io/reason": "RBAC: allowed by ClusterRoleBinding \"prometheus-k8s\" of ClusterRole \"prometheus-k8s\" to ServiceAccount \"prometheus-k8s/openshift-monitoring\""} }
```

- View the Kubernetes API server audit logs:
  - a. List the Kubernetes API server audit logs that are available for each control plane node:

```
$ oc adm node-logs --role=master --path=kube-apiserver/
```

### Example output

```
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit-2021-03-09T14-07-27.129.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit-2021-03-09T19-24-22.620.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit-2021-03-09T18-37-07.511.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit.log
```

- b. View a specific Kubernetes API server audit log by providing the node name and the log name:

```
$ oc adm node-logs <node_name> --path=kube-apiserver/<log_name>
```

For example:

```
$ oc adm node-logs ci-ln-m0wpfjb-f76d1-vnb5x-master-0 --path=kube-apiserver/audit-2021-03-09T14-07-27.129.log
```

### Example output

```
{ "kind": "Event", "apiVersion": "audit.k8s.io/v1", "level": "Metadata", "auditID": "cfce8a0b-b5f5-4365-8c9f-79c1227d10f9", "stage": "ResponseComplete", "requestURI": "/api/v1/namespaces/openshift-kube-scheduler/serviceaccounts/openshift-kube-scheduler-sa", "verb": "get", "user": { "username": "system:serviceaccount:openshift-kube-scheduler-operator:openshift-kube-scheduler-operator", "uid": "2574b041-f3c8-44e6-a057-baef7aa81516", "groups": ["system:serviceaccounts", "system:serviceaccounts:openshift-kube-scheduler-operator", "system:authenticated"], "sourceIPs": ["10.128.0.8"], "userAgent": "cluster-kube-scheduler-operator/v0.0.0 (linux/amd64) kubernetes/$Format", "objectRef": { "resource": "serviceaccounts", "namespace": "openshift-kube-scheduler", "name": "openshift-kube-scheduler-sa", "apiVersion": "v1"}, "responseStatus": { "metadata": {}, "code": 200, "requestReceivedTimestamp": "2021-03-08T18:06:42.512619Z", "stageTimestamp": "2021-03-08T18:06:42.516145Z", "annotations": { "authentication.k8s.io/legacy-token": "system:serviceaccount:openshift-kube-scheduler-operator:openshift-kube-scheduler-operator", "authorization.k8s.io/decision": "allow", "authorization.k8s.io/reason": "RBAC: allowed by ClusterRoleBinding \"system:openshift:operator:cluster-kube-scheduler-operator\" of ClusterRole \"cluster-admin\" to ServiceAccount \"openshift-kube-scheduler-operator/openshift-kube-scheduler-operator\""} }
```

- View the OpenShift OAuth API server audit logs:

- a. List the OpenShift OAuth API server audit logs that are available for each control plane node:

```
$ oc adm node-logs --role=master --path=oauth-apiserver/
```

### Example output

```
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit-2021-03-09T13-06-26.128.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit-2021-03-09T18-23-21.619.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit-2021-03-09T17-36-06.510.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit.log
```

- b. View a specific OpenShift OAuth API server audit log by providing the node name and the log name:

```
$ oc adm node-logs <node_name> --path=oauth-apiserver/<log_name>
```

For example:

```
$ oc adm node-logs ci-ln-m0wpfjb-f76d1-vnb5x-master-0 --path=oauth-apiserver/audit-2021-03-09T13-06-26.128.log
```

### Example output

```
{"kind":"Event","apiVersion":"audit.k8s.io/v1","level":"Metadata","auditID":"dd4c44e2-3ea1-4830-9ab7-c91a5f1388d6","stage":"ResponseComplete","requestURI":"/apis/user.openshift.io/v1/users/~","verb":"get","user":{"username":"system:serviceaccount:openshift-monitoring:prometheus-k8s","groups":["system:serviceaccounts","system:serviceaccounts:openshift-monitoring","system:authenticated"]},"sourceIPs":["10.0.32.4","10.128.0.1"],"userAgent":"dockerregistry/v0.0.0 (linux/amd64) kubernetes/$Format","objectRef":{"resource":"users","name":"~","apiGroup":"user.openshift.io","apiVersion":"v1"},"responseStatus":{"metadata":{"code":200},"requestReceivedTimestamp":"2021-03-08T17:47:43.653187Z","stageTimestamp":"2021-03-08T17:47:43.660187Z"},"annotations":{"authorization.k8s.io/decision":"allow","authorization.k8s.io/reason":"RBAC: allowed by ClusterRoleBinding \"basic-users\" of ClusterRole \"basic-user\" to Group \"system:authenticated\"}}
```

- View the OpenShift OAuth server audit logs:
  - a. List the OpenShift OAuth server audit logs that are available for each control plane node:

```
$ oc adm node-logs --role=master --path=oauth-server/
```

### Example output

```
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit-2022-05-11T18-57-32.395.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-0 audit.log
```

```
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit-2022-05-11T19-07-07.021.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-1 audit.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit-2022-05-11T19-06-51.844.log
ci-ln-m0wpfjb-f76d1-vnb5x-master-2 audit.log
```

- b. View a specific OpenShift OAuth server audit log by providing the node name and the log name:

```
$ oc adm node-logs <node_name> --path=oauth-server/<log_name>
```

For example:

```
$ oc adm node-logs ci-ln-m0wpfjb-f76d1-vnb5x-master-0 --path=oauth-server/audit-2022-05-11T18-57-32.395.log
```

### Example output

```
{"kind":"Event","apiVersion":"audit.k8s.io/v1","level":"Metadata","auditID":"13c20345-f33b-4b7d-b3b6-e7793f805621","stage":"ResponseComplete","requestURI":"/login","verb":"post","user":{"username":"system:anonymous","groups":["system:unauthenticated"],"sourceIPs":["10.128.2.6"],"userAgent":"Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0","responseStatus":{"metadata":{"code":302},"requestReceivedTimestamp":"2022-05-11T17:31:16.280155Z","stageTimestamp":"2022-05-11T17:31:16.297083Z","annotations":{"authentication.openshift.io/decision":"error","authentication.openshift.io/username":"kubeadmin","authorization.k8s.io/decision":"allow","authorization.k8s.io/reason":""}}}}
```

The possible values for the **authentication.openshift.io/decision** annotation are **allow**, **deny**, or **error**.

## 1.3. FILTERING AUDIT LOGS

You can use **jq** or another JSON parsing tool to filter the API server audit logs.



### NOTE

The amount of information logged to the API server audit logs is controlled by the audit log policy that is set.

The following procedure provides examples of using **jq** to filter audit logs on control plane node **node-1.example.com**. See the [jq Manual](#) for detailed information on using **jq**.

### Prerequisites

- You have access to the cluster as a user with the **dedicated-admin** role.
- You have installed **jq**.

### Procedure

- Filter OpenShift API server audit logs by user:

```
$ oc adm node-logs node-1.example.com \
  --path=openshift-apiserver/audit.log \
  | jq 'select(.user.username == "myusername")'
```

- Filter OpenShift API server audit logs by user agent:

```
$ oc adm node-logs node-1.example.com \
  --path=openshift-apiserver/audit.log \
  | jq 'select(.userAgent == "cluster-version-operator/v0.0.0 (linux/amd64)
  kubernetes/$Format")'
```

- Filter Kubernetes API server audit logs by a certain API version and only output the user agent:

```
$ oc adm node-logs node-1.example.com \
  --path=kube-apiserver/audit.log \
  | jq 'select(.requestURI | startswith("/apis/apiextensions.k8s.io/v1beta1")) | .userAgent'
```

- Filter OpenShift OAuth API server audit logs by excluding a verb:

```
$ oc adm node-logs node-1.example.com \
  --path=oauth-apiserver/audit.log \
  | jq 'select(.verb != "get")'
```

- Filter OpenShift OAuth server audit logs by events that identified a username and failed with an error:

```
$ oc adm node-logs node-1.example.com \
  --path=oauth-server/audit.log \
  | jq 'select(.annotations["authentication.openshift.io/username"] != null and
  .annotations["authentication.openshift.io/decision"] == "error")'
```

## 1.4. GATHERING AUDIT LOGS

You can use the `must-gather` tool to collect the audit logs for debugging your cluster, which you can review or send to Red Hat Support.



### NOTE

In OpenShift Dedicated deployments, customers who are not using the Customer Cloud Subscription (CCS) model must request a copy of your cluster's audit logs by contacting Red Hat Support. This is because using the `must-gather` tool requires **cluster-admin** privileges.

### Procedure

1. Run the **`oc adm must-gather`** command with **`-- /usr/bin/gather_audit_logs`**:

```
$ oc adm must-gather -- /usr/bin/gather_audit_logs
```

2. Create a compressed file from the **`must-gather`** directory that was just created in your working directory. For example, on a computer that uses a Linux operating system, run the following command:

```
$ tar cvaf must-gather.tar.gz must-gather.local.472290403699006248 1
```

**1** Replace **must-gather-local.472290403699006248** with the actual directory name.

3. Attach the compressed file to your support case on the [the Customer Support page](#) of the Red Hat Customer Portal.

## 1.5. ADDITIONAL RESOURCES

- [Must-gather tool](#)

## CHAPTER 2. REQUIRED ALLOWLIST IP ADDRESSES FOR SRE CLUSTER ACCESS

### 2.1. OVERVIEW

For Red Hat SREs to troubleshoot any issues within OpenShift Dedicated clusters, they must have ingress access to the API server through allowlist IP addresses.

### 2.2. OBTAINING ALLOWLISTED IP ADDRESSES

OpenShift Dedicated users can use an OpenShift Cluster Manager CLI command to obtain the most up-to-date allowlist IP addresses for the Red Hat machines that are necessary for SRE access to OpenShift Dedicated clusters.



#### NOTE

These allowlist IP addresses are not permanent and are subject to change. You must continuously review the API output for the most current allowlist IP addresses.

#### Prerequisites

- You installed the [OpenShift Cluster Manager API command-line interface \(ocm\)](#).
- You are able to configure your firewall to include the allowlist IP addresses.

#### Procedure

1. To get the current allowlist IP addresses needed for SRE access to your OpenShift Dedicated cluster, run the following command:

```
$ ocm get /api/clusters_mgmt/v1/trusted_ip_addresses|jq -r '.items[].id'
```

2. Configure your firewall to grant access to the allowlist IP addresses.